

Testseite für L^AT_EX

Die Umsetzung von Zetteltext nach L^AT_EX ist recht schwierig. Zum einen gibt es z.B. keine Standardisierung für ~~Durchgestrichen~~ und erst recht problematisch wird es bei Listen. Die normalen TAGs wie **fett**, *kursiv* und unterstrichen sollten dagegen gut funktionieren. Ein Beispiel: **fettunterstrichen**. Evtl. mach **fett mit kursiv + schiefgerade** ein paar Probleme. Allerdings sollten Super^{script} für z.B. EMC² und Sub_{script} für z.B. H₂O gut dargestellt werden. Sie sollten sich auch ohne Probleme in einen mehrzeiligen Text einfügen.

Manche UTF8-Sonderzeichen wie z.B. diese »Anführungszeichen« oder diese „Anführungszeichen“ kommen noch ganz gut rüber. Dagegen machen die standard “Anführungszeichen“ Probleme, sie müssen in englische “Anführungszeichen“ gewandelt werden. Schlimmer wird es allerdings schon mit © für Copyright[©] und ® für Trademarks[®]. Diese werden in .dvi Dateien nicht dargestellt, kommen jedoch in PDF-Dateien noch rüber.

* * * * *

Die Zeichen: % \$ & { } - # ! “ § () \? * + < > sind für L^AT_EX-Steuerzeichen und damit (möglicherweise) problematische Zeichen. Sie werden durch voranstellen von \bzw durch einpacken mit \$-Zeichen aber normal druckbar. Unter Umständen stimmen danach jedoch die Abstände nicht.

Es ist nicht möglich L^AT_EX-Code hier direkt einzugeben ... außer z.B. ⇒ ... der Pfeil sollte korrekt dargestellt werden.

In L^AT_EX hat der »\« eine besondere Bedeutung; so wird mittels \begin{verbatim} beispielsweise der Ausdruck von Listings eingeleitet.

Listing Beispiel:

```
void pop_zindex ()
{
    int i;
    GtkWidget *button;

//  g_printf("+++ in pop_zindex");
    for (i=LEN_ZSTACK; i>0; i--) {
        zstack[i] = zstack[i-1];
//  g_print(" %d", zstack[i]);
    }
    zstack[i] = 0;
    zindex = zstack[LEN_ZSTACK];
    if (zstack[LEN_ZSTACK-1] == 0) {
        button = lookup_widget (GTK_WIDGET (main_window), "main_btn_undo");
        gtk_widget_set_sensitive(button, FALSE);
    }
//  g_printf("\n");
}
```

Die nachfolgenden Sonderzeichen sind mit L^AT_EX nicht (direkt) darstellbar ... sind aber möglicherweise sinnvolle Zeichen:

© ® ⇒ ✓